#65   PATENT

1.   (Currently amended) A method for identifying and removing given content files ~~from a set of content servers in a content delivery network, wherein content servers share content files with each other~~, comprising:

identifying content files to be removed from ~~the content servers~~ <u>a set, or a given subset, of</u>
5   <u>content servers in a content delivery network, wherein the content delivery network is operated</u>
<u>by a service provider on behalf of participating content providers who use the content servers to</u>
<u>cache and serve content files, and wherein the content servers in the content delivery network</u>
<u>share content files with each other</u>;

pushing an aggregate purge request from <u>a central server</u> to each of a set of staging
10 servers, each aggregate purge request including an identifier for each content file to be removed
from the <u>set, or the given subset, of</u> content servers;

periodically, having each of the set, <u>or the given subset,</u> of content servers obtain the
aggregate purge request from a given staging server, wherein the content servers obtain the
aggregate purge request independently and at different times; and

15   at each content server <u>that obtains the aggregate purge request</u>, purging from the content
server each content file identified in the aggregate purge request;

wherein after a first content server ~~in the set of content servers~~ has executed the aggregate
purge request and, as a result, has purged a given content file, inhibiting the first content server
from receiving the given content file from a second content server ~~in the set of content servers~~
20 with which the first content server shares content files if the second content server has <u>received</u>
<u>but</u> not then executed the aggregate purge request.

2

#65   PATENT

2.   (Original) The method as described in Claim 1 wherein the aggregate purge request is pushed to the each of the set of staging servers over a secure link.

3.   (Original) The method as describe in Claim 1 wherein the aggregate purge request is pulled from the given staging server to each of the set of content servers over a secure link.

5   4.   (Original) The method as described in Claim 1 further including the step of issuing a notification that each content file identified in the aggregate purge request has been purged from the content delivery network.

5.   (Original) The method as described in Claim 1 further including the step of issuing a notification that each content file identified in the aggregate purge request has been
10   accepted for purging.

6.   (Previously presented) The method as described in Claim 1 wherein the step of identifying the content files to be removed from the content servers includes the step of verifying that a user requesting removal is authorized to purge the content files.

7.   (Original) The method as described in Claim 6 wherein the user is a content
15   delivery network customer.

8.   (Original) The method as described in Claim 6 wherein the user is a content delivery network administrator.

9.   (Cancelled).

10.   (Cancelled).

20

3

#65   **PATENT**

11.   (Currently amended) ~~In~~ In a content delivery network ~~wherein third party content is cached on and served from a set of content servers in response to end user requests,~~ operated by a service provider on behalf of participating content providers, wherein the participating content providers use content servers in the content delivery network to cache and serve content files, and

5   wherein content servers in the content delivery network share content files with each other, the improvement comprising:

   a purge mechanism for selectively identifying and removing given content files from the set, or a given subset, of content servers in the content delivery network, comprising:

   a Web-based interface for identifying content files to be purged from the content

10   servers;

   a purge server for receiving purge requests pushed from the Web-based interface, validating each purge request, and batching a set of purge requests into an aggregate purge request;

   a set of staging servers for receiving the aggregate purge request pushed from the

15   purge server; and

   code executing on a given content server for periodically polling a given staging server, for pulling the aggregate purge request, for removing the identified content files from the given content server, and, with respect to a given content file that has been removed from the given content server, for inhibiting the given content server from

20   receiving the given content file from at least one other content server with which the given content server shares content files if the other content server has not then removed the given content file that has been removed by the given content server.

   12.   (Cancelled).

25

4